

CLAIMS

What is claimed is:

1. A computer implemented method for remote access to files for a local agent module, comprising:

5 polling a server for a task request;
receiving a task request from the server;
executing a task from the task request;
causing a file to be uploaded, the file identified in the task request, to a server;
10 waiting for a schedule timer to expire; and
repeating the above acts, beginning with the act of polling.

2. The method of claim 1, further comprising:

15 setting up local agent preferences;
setting up remote client preferences;
initiating the act of polling, based on the local agent preferences; and
initiating an act of uploading, based on the remote
20 client preferences.

3. The method of claim 1, wherein the act of polling occurs over a transmission control protocol/internet protocol stack, through functions specified in a simple
25 object access protocol interpreter.

4. The method of claim 1, wherein the act of executing the task includes:

30 initiating a request to a subsystem for the file; and
receiving the file from the subsystem.

5. The method of claim 1, wherein the act of executing the task includes:

initiating a request to a subsystem for the file;

5 instructing the subsystem to upload the file to the server; and

 receiving an indication that the file was uploaded to the server.

10 6. The method of claim 1, wherein the act of executing the task includes:

 initiating a request to a message access protocol interface for the file from a message access protocol interface database; and

15 receiving the file from the message access protocol database.

20 7. The method of claim 6, wherein the act of uploading the file includes instructing the file to be sent to the server from the message access protocol database.

8. A computer readable medium including sequences of instructions for causing one or more processors to perform acts for remote file access for a local agent module, the sequences of instructions comprising:

polling a server for a task request;

receiving a task request from the server;

executing a task from the task request;

causing a file to be uploaded, the file identified in

30 the task request, to a server;

waiting for a schedule timer to expire; and
repeating the above acts, beginning with the act of
polling.

5 9. The computer readable medium of claim 8, further
comprising instructions for performing the acts of:
setting up local agent preferences;
setting up remote client preferences;
initiating the act of polling, based on the local agent
10 preferences; and
initiating an act of uploading, based on the remote
client preferences.

10 10. The computer readable medium of claim 8, wherein the
15 act of polling occurs over a transmission control
protocol/internet protocol stack, through functions
specified in a simple object access protocol interpreter.

20 11. The computer readable medium of claim 8, wherein the
act of executing the task includes:
initiating a request to a subsystem for the file; and
receiving the file from the subsystem.

25 12. The computer readable medium of claim 8, wherein the
act of executing the task includes:
initiating a request to a subsystem for the file;
instructing the subsystem to upload the file to the
server; and
30 receiving an indication that the file was uploaded to
the server.

13. The computer readable medium of claim 8, wherein the act of executing the task includes:

initiating a request to a message access protocol interface for the file from a message access protocol interface database; and

receiving the file from the message access protocol database.

14. The computer readable medium of claim 13, wherein the act of uploading the file includes instructing the file to be sent to the server from the message access protocol database.

15. A local agent comprising:

a task processor for polling a server for a task request, the task request identifying a file in a local computer;

a schedule timer communicatively coupled to said task processor for controlling a task processor polling interval; and

one or more protocol stacks for communicating over a network with the server.

16. The local agent of claim 15, wherein the one or more protocol stacks includes a transmission control protocol/internet protocol stack.

17. The local agent of claim 15, wherein the one or more protocol stacks includes a simple object access protocol interpreter.

18. The local agent of claim 15, further comprising a subsystem for executing a task from the task request.

5 19. The local agent of claim 15, further configured to initiate a request to a message application programming interface database.

10 20. The local agent of claim 15, further configured to receive a file from a message application programming interface database.

21. A local agent module for remote access to files, comprising:

15 a transmission control protocol/internet protocol stack for network communication with a server over a network;

an extensible markup language input/output parser, communicatively coupled to the transmission control/internet protocol stack, for breaking down data and
20 commands;

a simple object access protocol interpreter, communicatively coupled to the extensible markup language input/output parser, for creating file system instructions to poll the server for a task request and retrieve a file
25 specified in the task request;

and a task processor, communicatively coupled to the simple object access protocol interpreter, for executing subsystem instructions and initiating poll commands, based on a schedule timer.

30

22. The local agent module of claim 21, wherein local agent module includes a communications module configured to provide a carrier for network communication to the server, the local agent module configured to periodically connect
5 to the server through the communication module at intervals set by the schedule timer.

23. The local agent module of claim 22, wherein the local agent module resides in a memory of a local computer, and
10 is configured to access files in a storage device associated with the local computer.

24. The local agent module of claim 21, further comprising a message application programming interface,
15 communicatively coupled to the task processor, for allowing access to a message application protocol interface database.

2025-10-20 10:45:00